

Association for Information Systems AIS Electronic Library (AISeL)

ECIS 2014 Proceedings

A PROPOSAL FOR USING WEB 2.0 TECHNOLOGIES IN SCRUM

Irapuan Glória Júnior

Universidade Nove de Julho, Sao Paulo, Brazil, ijunior@ndsgn.com.br

Rodrigo Oliveira

Universidade Nove de Julho, Sao Paulo, Brazil, rodrigo.oliveira@tivit.com.br

Marcirio Chaves

Universidade Nove de Julho, Sao Paulo, Brazil, mschaves@gmail.com

Follow this and additional works at: <http://aisel.aisnet.org/ecis2014>

Irapuan Glória Júnior, Rodrigo Oliveira, and Marcirio Chaves, 2014, "A PROPOSAL FOR USING WEB 2.0 TECHNOLOGIES IN SCRUM", Proceedings of the European Conference on Information Systems (ECIS) 2014, Tel Aviv, Israel, June 9-11, 2014, ISBN 978-0-9915567-0-0
<http://aisel.aisnet.org/ecis2014/proceedings/track13/4>

This material is brought to you by the European Conference on Information Systems (ECIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in ECIS 2014 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

A PROPOSAL FOR USING WEB 2.0 TECHNOLOGIES IN SCRUM

Irapuan Glória Júnior, Master of Project Management Program, University Nove de Julho (UNINOVE), Av. Francisco Matarazzo, 612, Água Branca 05001-000, São Paulo, SP, Brazil, ijunior@ndsgn.com.br

Rodrigo Oliveira, Master of Project Management Program, University Nove de Julho (UNINOVE), Av. Francisco Matarazzo, 612, Água Branca 05001-000, São Paulo, SP, Brazil, rodrigo.oliveira@tivit.com.br

Marcirio Silveira Chaves, Master of Project Management Program, University Nove de Julho (UNINOVE), Av. Francisco Matarazzo, 612, Água Branca 05001-000, São Paulo, SP, Brazil, mschaves@uninove.br

Abstract

One of the challenges in Project Management (PM) is the introduction of information technology in PM methodologies. To deal with this issue, the use of Web 2.0 technologies to support virtual work and virtual management of projects has gained attention in recent years. At the same time, the adoption of agile methodologies has grown to better assist mainly the management of information technology projects. This paper proposes a conceptual contribution through the use of Web 2.0 technologies in Scrum methodology. It suggests Web 2.0 tools to fit each artifact and event Scrum, which provide an environment for virtual management of projects. Such tools support meetings and knowledge sharing. It also presents the research questions (RQ) and methods that will be used to conduct this work. RQ1 is “How can Web 2.0 technologies be integrated into Scrum methodology to monitor projects and store and share knowledge generated in the meetings?” and RQ2 is “To what extent do Web 2.0 technologies improve the performance of Scrum teams?” To address RQ1 and RQ2, we will use multiple case study and action research methods, respectively.

Keywords: IT 2.0, Project Management, Agile Methodologies, Scrum.

1 Introduction

Emerging Web 2.0 technologies and applications have contributed to virtual work in organizations. They are also starting to gain visibility and use by project managers to better support daily tasks and processes (Boulos, Maramba and Wheeler, 2006; Cleveland, 2012; Grace, 2009; Shang, Li, Wu and Hou, 2011). Chi (2008) describes how project managers are using Web 2.0 technologies such as blogs and Rich Site Summary (RSS). Gholami and Murugesan (2011) detail how the managers of global IT projects are using Web 2.0 technologies to support everyday tasks and thus improve the management of a project as a whole. IT projects are becoming increasingly complex and thus create the need for business models to develop software to meet the new demands (Hui, 2010). In this context, agile methodologies such as Extreme Programming (Beck and Andres, 1999), Crystal (Brun, Notkin, Holmes and Ernst, 2013) and Scrum (Schwaber and Sutherland, 2011) have emerged to support the development of software. The use of new technologies in agile methodologies can facilitate the faster delivery of IT projects, such as the application of Web 2.0 technologies, which in addition to collaboration, boasts immediacy as its main feature (O'Reilly, 2005).

Specifically, research in Scrum has been focused on: (1) The application of dynamic scoping projects and the use of shorter sprints to keep up with changes (Pereira, Turret, and Marcal, 2007), (2) failures in communication between members of the Scrum Team (Hummel, Rosenkranz and Holten, 2013), (3) application of CMMi methodology in project development (Zanatta and Vilain, 2005), (4) the use of Scrum in various types of organizational environments (Rayhan and Haque, 2008; Moe, Dingsoyr and Dyba, 2010), and (5) the creation of a new methodology joining The Open Group Architecture Framework (TOGAF) and Scrum (Buckl, Matthes, Monahov, Roth, Schulz, and Schweda, 2011).

Despite the wealth of information in agile methodologies and project management, we cannot find works integrating Web 2.0 technologies and every phase of Scrum. This paper presents a contribution to the field of project management through a proposal for integrating Web 2.0 tools in Scrum artifacts and events. This proposal could facilitate the virtual work and the virtual management of projects.

2 Theoretical Background

2.1 Web 2.0 Technologies

Over the past few years, the World Wide Web has undergone numerous changes, leading to the departure of Web 1.0 to Web 2.0, a term adopted by Tim O'Reilly and MediaLive International in 2004 (O'Reilly, 2005; Levy, 2009). Sharing and content creation has become a great ally to change, causing users to experience writing web pages through the new platforms that have gained strength such as Wikis and blogs (Gould, 2009; Click and Petit, 2010). Web-based Office Suite is a platform that allows collaborative editing, another relevant feature in the context of a project.

Real-time text and voice communication allows interaction between people in the web. Web-based Office Suites such as Google Drive offers chat built-in the platform. A blog is a discussion or informational site published on the Web which consists of entries displayed in reverse chronological order. Most blogs are interactive (i.e. open to comments by visitors) and have a set of characteristics which allow them to gain popularity. In the context of a project, a reverse blog, which is composed by its users, can be useful. Reverse blogs can be used for tasks such as project progress reporting that managers or clients can easily see.

Rich Site Summary (RSS) is a set of web feed formats used to publish frequently updated works such as wiki and blog entries, audio and video in a standardized format. Using this technology, project

members can be automatically notified of updates in a wiki or a blog used in a project, eliminating the need of periodic visits to search for updates in these sites.

2.2 Agile Methodologies

Agile methodologies are based on the agile manifesto, which promises to deliver greater productivity, quality and chance of success in software development projects. One of the features is to consider the environment as dynamic, with predicted and expected changes (Beck et al., 2001). Widely used Agile methodologies include Agile Extreme Programming, Crystal and Scrum.

Extreme Programming is characterized as a distinct pair programming, where one programmer performs coding and another verifies the code simultaneously (Andres and Beck, 1999). The Crystal agile methodology can be applied to teams of up to 8 developers. It focuses on efficiency, contributing to the success of the project. (1) Crystal Clear has the following properties: (2) It focuses on people, not processes or artifacts, (3) Deliverables contained in the code, (4) It can generate personal security to staff members, (5) Frequent integration (Highsmith, 2002; Cockburn, 2004). Scrum is applied in development projects with small teams, which makes it attractive to most companies. The use of small development cycles allows changes in volatile environments to help the system adapt more quickly to respond to the changes (Schwaber and Sutherland, 2011). According to a survey described in West, Grant, Gerushand and D'Silva (2010), Scrum is the most cited agile method in use.

2.2.1 Scrum

Scrum is a framework for developing and maintaining complex products through roles, events, artifacts and rules that keep them integrated. A new perspective, Scrum, arising from self-organizing teams, promotes the breakdown of the traditional model of determinism, auditability and control (Moe, Dingsoyr and Dyba, 2010). The self-organization of a team member is closely associated with the type of organizational culture (Rayhan and Haque, 2008). Using Scrum promotes other benefits, such as the satisfaction of team members, low turnover and absenteeism (Moe, Dingsoyr and Dyba, 2010).

Scrum Artifacts, Roles and Events

Sprint is a cycle of up to one month of planning tasks. The end of a Sprint indicates a new version of the product, called "Done" (Schwaber and Sutherland, 2011). We should also note that there are teams that have a better outcome when each Sprint is restricted to one week (Moe, Dingsoyr and Dyba, 2010). The *Product Backlog* is a list of requirements for product development. It is anticipated that there are changes in business requirements, market conditions or technology that may affect the settings made (Rayhan and Haque, 2008). The *Sprint Backlog* is the prediction of the development team regarding what features will be available in the next version of the project. It corresponds to a part of the *Product Backlog* (Schwaber and Sutherland, 2011).

The *Scrum Team* is composed of the *Product Owner*, the *Development Team* and *Scrum Master*. The assignments are defined before the start of the project so that no communication problem occurs (Rayhan and Haque, 2008). The *Product Owner* is responsible for maximizing the value of the product and the work of the Development Team by elucidating questions or the determination of product features. The *Development Team*, composed of three to nine members of various profiles, is responsible for performing deliberate actions established by the *Product Owner*. The *Scrum Master* is responsible for ensuring that Scrum is understood and applied by all involved, from the *Development Team*, *Product Owner* to those who will interact with the *Scrum Team* (Schwaber and Sutherland, 2011).

Events in Scrum aim to create a routine and restrict meetings to those specified. The framework of a Sprint consists of: (1) The Sprint Planning Meeting, (2) Daily meetings, (3) Work team development, (4) Sprint Review and (5) Sprint Retrospective (Schwaber and Sutherland, 2011).

The Sprint Planning Meeting is created by the collaborative work of the entire Scrum Team, based on what was defined in the Product Backlog. The presence at the Sprint Planning Meeting of the Product Owner is required to clarify the Product Backlog items, select conflicting decisions and assist in the exchange, such as the excess or lack of work (Schwaber and Sutherland, 2011).

The Daily Scrum is a 15-minute time-boxed event for the Development Team to synchronize activities and create a plan for the next 24 hours. This is done by inspecting the work carried out since the last Daily Scrum and forecasting the work that could be done before the next one (Schwaber and Sutherland, 2011).

A Sprint Review is held at the end of the Sprint to inspect the Increment and adapt the Product Backlog if needed. During the Sprint Review, the Scrum Team and stakeholders collaborate on what was done in the Sprint (Schwaber and Sutherland, 2011).

Finally, the Sprint Retrospective is an opportunity for the Scrum Team to inspect itself and create a plan for improvements to be enacted during the next Sprint (Schwaber and Sutherland, 2011). In Project Management this time is termed as the record of lessons learned (PMI, 2012).

3 A Proposal for Using Web 2.0 Technologies in Scrum

The deployment of Web 2.0 technologies in Scrum methodology requires that users have clear guidance on how they can handle these technologies and in what situations they occur in each phase of Scrum. Table 1 summarizes the recommended tools for each phase, and Figure 1 outlines the integration of these tools in the Scrum framework.

Initially the *Product Backlog* is created in the collaborative editing environment, where it can be changed in real time with stakeholders. A tool like Google Drive allows collaborative editing and editing control through specific profiles (Google, 2014). Besides the administrator profile of the *Product Owner*, it is possible to create a second profile with access to the content restricted to reading only by certain stakeholders. Then, in creating *Sprint Backlog* the team and the *Product Owner* will select the tasks that will be moved to the new folder. This new folder should be named with the version number corresponding to the date of their creation, e.g. Sprint_0001_20131208. This folder should be stored in a collaborative editing tool, such as Google Drive (Google, 2014). Real Time Collaborative Editors provide a more complete list of such tools.

In the *Sprint's Planning* the team will assign tasks to members and their delivery dates. We suggest the use of a blog tool implementing reverse blog, such as Blogger (<http://blogger.com>). Daily meetings will take place using a VoIP tool such as Skype (<http://www.skype.com>), Google Talk (<http://www.google.com/hangouts>) or Viber (<http://www.viber.com>), where the log will store communication via chat and will serve as the minutes of the meeting. The head of the meeting should: a) purify this file, b) store it on the collaborative editing platform in a folder called "MINUTE", for example, and c) rename the label with the prefix "MINUTE" followed by an underscore and the date of the meeting format YYYYMMDD, an underline and two or three words on the theme of the meeting. For example, "MINUTE_20131120_SolutionsPerformanceUpload", which is understood as the minutes of the meeting held on 11/20/2013, on the performance upload solution.

During product development, communication between team members must be made via internal microblogging, such as Yammer (<http://www.yammer.com>). The team may request assistance from other members, post development tips and sharing status within other relevant information. The *Sprint*

Review is done at the end of the sprint, and will report via blog the status of each action in the blog's initial sprint. Tools for managing blogs include BlogJet (<http://www.blogjet.com>), Veeeb (<http://www.veeb.com>) and Blogger (<http://blogger.com>). Hongkiat (2013) presents a more comprehensive list of this kind of tool. Finally, before the next cycle, the *Sprint's Retrospective*, a staff member will be responsible for documenting lessons learned throughout the sprint in the Wiki environment so that other members can consult at any time. In sum, Web 2.0 technologies proposed the integration of the Scrum framework and the introduction of new forms of conferencing and interacting. They cover all Scrum events and artifacts and facilitate virtual work.

Scrum Phase/Artifact/Event	Solution
Product Backlog	Collaborative Editing
Sprint Backlog	Collaborative Editing
Sprint's Planning	Reverse Blog
Daily Meeting	Communication Text/Voice and Collaborative Editing
Team Development	Microblogging
Sprint's Review	Blog
Sprint's Retrospective	Wiki

Table 1. Technology-based solutions applied to Scrum artifacts and events.

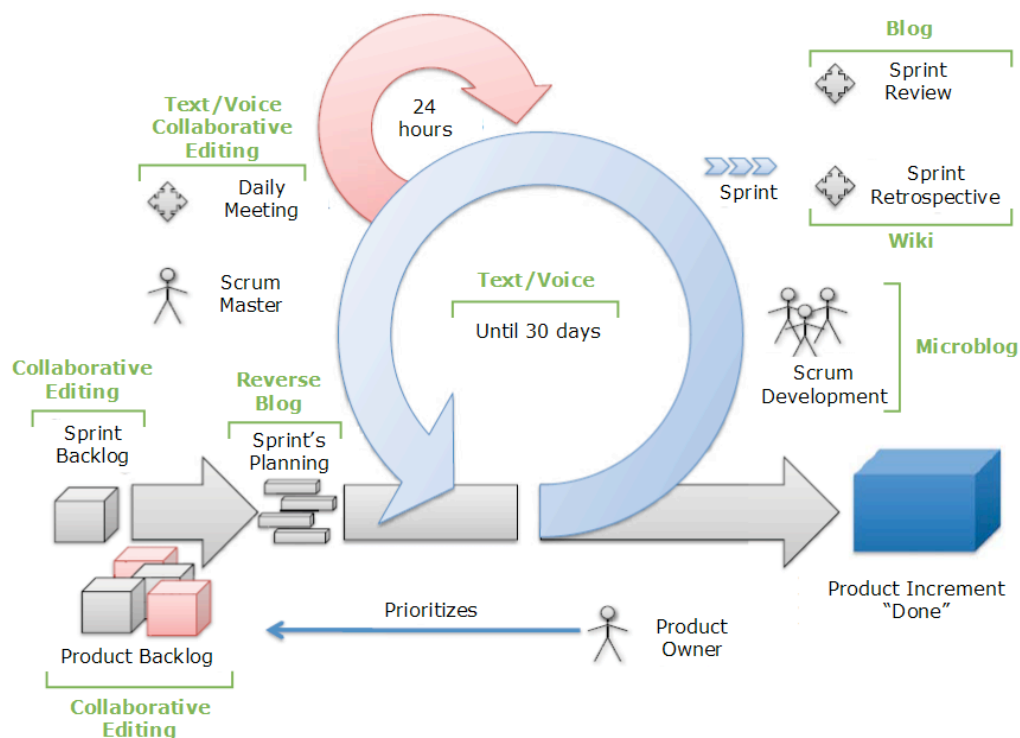


Figure 1. A proposal for using Web 2.0 technologies in Scrum. Based in Buckl et al. (2011).

4 Research Questions and Discussion

We will validate the proposal described in this paper in organizations. For this, we will use multiple case study and action-research methods to answer the following research questions, respectively:

RQ1: How can Web 2.0 technologies be integrated into the Scrum methodology to monitor projects and store and share knowledge generated in the meetings?

RQ2: To what extent do Web 2.0 technologies improve the performance of Scrum teams?

To address RQ1, we will conduct a multiple case study in organizations already using Scrum and with a culture based on agile methods. We will carry out interviews with Project Managers and a survey with the Scrum team. Moreover, we will analyze the documents provided by the companies. The result analysis will be based on the triangulation of the data.

To address RQ2, we will use the action research method which is appropriate for dealing with Scrum artifacts and events. Figure 2 presents the design of the research with the application of action research to Scrum. Initially, the Scrum team creates the *Product Backlog*. The first cycle of action research starts with the *Sprint Backlog* followed by *Sprint Planning*, *Daily Meetings* and *Team Development* correspond to the Taking Action phase of the action research. To close this cycle, an evaluation is carried out in the *Sprint Review* and *Sprint Retrospective*. The analyzing and diagnosing phase of the new cycle will use the results of this evaluation. It is expected that as more cycles are performed the tasks become part of the culture of the team and thus the speed to complete the model is faster and more effective (Rayhan and Haque, 2008).

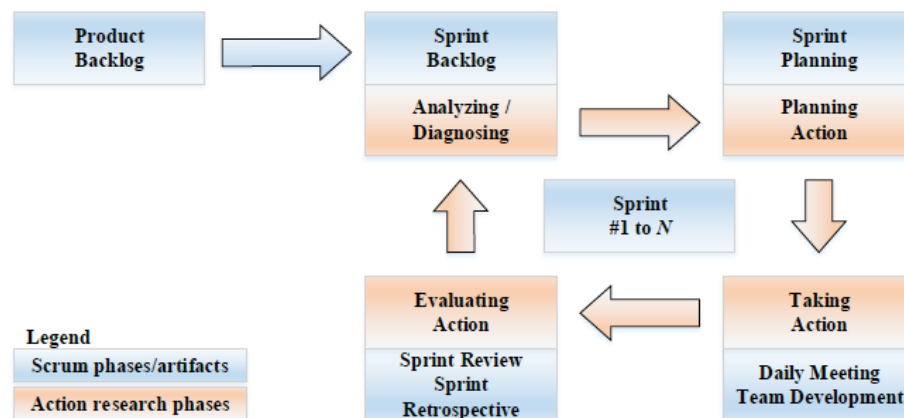


Figure 2. Design of the research with the application of action research to Scrum.

5 Final Remarks

This paper presents a proposal for using Web 2.0 technologies in Scrum. It covers all Scrum events and artifacts and highlights the possibility of using these technologies to facilitate virtual work. This proposal needs to be developed through its validation in organizations. A multiple case study followed by an action research seems to be an appropriate approach to deal with this problem. The objective of participating in this conference is to obtain feedback from the community before testing the proposal in the field. Finally, as far as we know, this study is the first to present a proposal for integrating Web 2.0 tools in Scrum artifacts and events.

References

- Beck, K. and Andres, C. (1999). *Extreme Programming Explained*, Addison-Wesley; 2nd edition.
- Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R.C., Mellor, S., Schwaber, K., Sutherland, J. and Thomas, D. (2001). *Agile Manifesto*. <http://www.agilemanifesto.org/>.

- Boulos, M., Maramba, I. and Wheeler, S. (2006). "Wikis, blogs and podcasts: a new generation of Web-based tools for virtual collaborative clinical practice and education." *BMC Medical Education* 6.1: 41.
- Brun, Y., Notkin, D., Holmes, R. and Ernst, M. D. (2013). Early Detection of Collaboration Conflicts and Risks. *IEEE Transactions on Software Engineering*, vol. 10, 1358-1375.
- Buckl, S., Matthes, F., Monahov, I., Roth, S., Schulz, C. and Schweda, C. M. (2011). Towards an Agile Design of the Enterprise Architecture Management Function, *IEEE*, 322-329.
- Chi, E. H. (2008). The Social Web: Research and Opportunities. *Computer*, 41(9), 88–91.
- Cleveland, S. (2012). Using Microblogging for Lessons Learned in Information Systems Projects. In *International Research Workshop on IT Project Management*, 122–128.
- Click, A. and Petit, J. (2010). Social Networking and Web 2.0 in Information Literacy. *The International Information & Library Review*, 137–142.
- Cockburn (2004) A. *Crystal clear: a human-powered methodology for small teams*. Addison-Wesley Professional. 336 pages.
- Gholami, B. and Murugesan, S. (2011). "Global IT Project Management Using Web 2.0." *International Journal of Information Technology Project Management (IJITPM)*, 2(3), 30-52.
- Google (2014). Google Inc. Available at <http://drive.google.com>. Last access 2014/03/13.
- Gould (2009). L.S. What Web 2.0 means to you. *Manage*, 36-37.
- Grace, T. P. L. (2009). Wikis as a knowledge management tool. *Journal of Knowledge Management*, 13(4), 64-74.
- Highsmith, J. A. (2002). *Agile software development ecosystems* (Vol. 13). Addison-Wesley Professional.
- Hongkiat (2013). Hongkiat.com - Design, Inspiration and Tecnology. Available at <http://www.hongkiat.com/blog/blogging-application-tools/> Last visit November, 2013.
- Hui, F. (2010). Research on the Developmental Management Model of Software Type IT Project, *IEEE*, 1-4.
- Hummel, M., Rosenkranz, C. and Holten, R. (2013), Explaining Changing Communication Paradigma of Agile Information System Development: a Reserch Model, Measurement Development and Pretest, *European Conference on Information Systems (ECIS)*, 1-12.
- Levy, M. (2007). Web 2.0 implications on knowledge management. *Journal of Knowledge Management*, 13(1), 120–134.
- Moe, N B, Dingsoyr, T. and Dyba, T. (2010). A temawork model for understanding an agile team: A case of a Scrum project, *Information and Software Technology*, number 52, 480-491.
- O'Reilly, T. (2005). What Is web 2.0: Design Patterns and Business Models for the Next Generation of Software. *Communications & Strategies*, 17–37.
- Pereira, P., Torreão, P. and Marçal, A. S. (2007). Entendendo Scrum para Gerenciar Projetos de Forma Ágil. *Mundo PM*, 1, 3–11.
- PMI (2012). *A Guide to the Project Management Body of Knowledge (PMBOK® guide)*. 5th ed., Project Management Institute. Newtown Square, PA., ISBN: 978-1935589-67-9, pp: 410.
- Rayhan, S. and Haque, N. (2008). Incremental Adoption of Scrum for Sucessful Delivery of an IT Project in a Remote Setup, *Agile 2008 Conference - IEEE*, 351-355.
- Schwaber, K. and Sutherland, J. (2013). *The Scrum Guide - The Definitive Guide to Scrum: The Rules of the Game*, Scrum Org.
- Shang, S. S. C., Li, E. Y., Wu, Y.-L. and Hou, O. C. L. (2011). Understanding Web 2.0 service models: A knowledge-creating perspective. *Information & Management*, 48(4-5), 178–184.
- West, D., Grant, T., Gerush, M. and D'Silva, D. (2010) Agile development: Mainstream adoption has changed agility. *Forrester Research*. Wiley Publishing, number 22, 435–455.
- Zanatta, A. L. and Vilain, P. (2005). Uma análise do método ágil Scrum conforme abordagem nas áreas de processo Gerenciamento e Desenvolvimento de Requisitos do CMMI. In *WER*, 209–220.